Atty Docket No. A2003009(2)
Kottke, et al.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

RECEIVED
CENTRAL FAX CENTER
APR 1 8 2008

In re Application of:	§	
	§	
Avid Technology, Inc.	§	
	§	
Serial No.: 10/817,217	§	Examiner: Jessica M. Roberts
	§	
Filed: April 2, 2004	§	
	§	
Title: FIXED BIT RATE, INTRAFRAME COMPRESSION AND DECOMPRESSION OF VIDEO	§	
	§	
	§	
	§	

RULE 131 DECLARATION OF DANE P. KOTTKE

The undersigned declarant, a co-inventor/co-applicant in the above identified application states:

1. I am Senior Principal Software Engineer of Avid Technology, Inc. ("Avid".) I am a co-inventor of the subject matter presently claimed in the above-identified patent application. I have been employed by Avid since 1999. My duties at Avid include the research and development of video codec technology. The matters set forth herein are based on my personal knowledge or reference to company records.

2. I have reviewed the above application and the amended claim before the Examiner, the outstanding PTO Office Action dated 19 February 2008, the patent application and patents used in rejecting the pending patent claims, and state the following regarding the invention of the present application, in relation to asserted reference US Patent No. 7,212,681 ("Chen"), which was filed 13 January 2003.

3. The present invention as described and claimed comprises a method for fixed bit

rate, intraframe compression of video, including a sequence of images. The instant patent application claims priority to US Provisional Patent Application No. 60/460,517 filed 04 April 2003.

4. I and my co-inventors conceived of the invention prior to the 13 January 2003 effective date of the Chen reference, and used reasonable diligence before that date to reduce the claimed invention to practice. Provided as Exhibit A hereto are copied portions of an Avid design document created in 2002, prior to the effective date of the Chen reference, that demonstrate our conception of the presently claimed invention prior to the effective date of the Chen reference.

5. I and my co-inventors worked diligently to reduce the invention to practice, with the first public use of an implementation of the invention, the Avid DNxHD codec (http://www.avid.com/dnxhd/), occurring in early 2004.

6. The Avid DNxHD codec is the first codec compliant with the recently published Society of Motion Picture and Television Engineers VC-3 standard.

7. The Avid DNxHD codec has enjoyed significant commercial success. To date, over 30 companies not owned or controlled by Avid have signed license agreements with Avid related to the use of the codec.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both under Section 1001 of Title 18 of the United State Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Respectfully submitted,

Dated: April 17, 2008

By: Dane P. Kottke
Dane P. Kottke

EXHIBIT A
(4 pages)

Avid Confidential

Avid.

Avid HD Variable Length Coding Description

Avid Technology
Revision: 00.04.00
Author: Dane P. Kottke

Date: [redacted]

RECEIVED
CENTRAL FAX CENTER
APR 1 8 2008

Table 2 VLC Efficiency at Various Bit-rates (8bit/pixel data)

Apply at \ Design at	218 Mbit/s	60 Mbit/s
218 Mbit/s	5.20 bits/symbol	5.49 bits/symbol
60 Mbit/s	5.14 bits/symbol	4.85 bits/symbol

Table 3 VLC Efficiency at Various Bit-rates (10bit/pixel data - extrapolated)

Apply at \ Design at	218 Mbit/s	60 Mbit/s
218 Mbit/s	5.22 bits/symbol	5.52 bits/symbol
60 Mbit/s	5.27 bits/symbol	5.03 bits/symbol

msb ... lsb

nbf	0	0	a	a_3	a_2	a_1	a_0	r_3	r_2	r_1	r_0	n_3	n_2	n_1	n_0

Figure 6 Format of VLD Table for VLDLUT Operation

3.4 Variable Length Decoding Table Format

The format of the decoding table is shown in Figure 6. The first bit is the nbf (Not Base Flag). When this bit is set, it indicates that the codeword does not originate from the base table codewords (i.e., V^b). The next two bits are not used and are arbitrarily set to zero. The next four bits encode the coefficients amplitude. The following four encode the run-length. The final four indicate the codeword's bit-length. Thus, each entry in the table can be stored in two bytes. If all the entries are stored in a single table, the total memory requirements are $2^{15}\text{entries} * 2\frac{\text{bytes}}{\text{entry}} = 65536\text{ bytes}$. Note that to save memory its possible to break the table into two or more tables. The reduced size is on the order of 16,384 bytes.

4. Alternating Run and Amplitude Variable Length Coding Description

This design partitions the range of amplitudes into two ranges: a base range for amplitudes between 1 and 64 and an index range for amplitudes between 65 and 4096. Amplitudes in the base range are encoded with a Huffman codeword. The index range is broken down into a number of segments, each of length 64. Amplitudes in the index range are encoded with a Huffman codewordand an index value that indicates from which segment they originate.

Coefficients with no preceding run of zeros are considered a different type of symbol from those that have one or more preceding zero-valued coefficients. If there is one or more preceding zero valued coefficients, the amplitude is encoded by a Huffman codeword (and, possibly an index value) and followed by another Huffman codeword

representing the length of the preceding run of zeros. If there is no preceding run of zeros, the amplitude is encoded with a Huffman codeword. A more precise description is given in the subsections below.

4.1 Symbol Definitions

There are six types of symbol sets: four for amplitude symbols, one for run lengths, and one for End of Block:

4. $A^{nrb} = \{A_1^{nrb}, A_2^{nrb}, \ldots, A_{64}^{nrb}\}$: Non-zero amplitude coefficients in the base range, with *no* preceding run of zero valued coefficients. The amplitudes vary from $A_1^{nrb} = 1$ to $A_{64}^{nrb} = 64$.
5. $A^{wrb} = \{A_1^{wrb}, A_2^{wrb}, \ldots, A_{64}^{wrb}\}$; Non-zero amplitude coefficients in the base range, *with* preceding run of zero valued coefficients. The amplitudes vary from $A_1^{wrb} = 1$ to $A_{64}^{wrb} = 64$.
6. $A^{nri} = \{A_1^{nri}, A_2^{nri}, \ldots, A_{64}^{nri}\}$: Non-zero amplitude coefficients in the index range, with *no* preceding run of zero valued coefficients. The amplitudes vary from 65 to 4096. The mapping from amplitude to symbol is explained in Section 4.1.1.
7. $A^{wri} = \{A_1^{wri}, A_2^{wri}, \ldots, A_{64}^{wri}\}$: Non-zero amplitude coefficients in the index range, with *with* preceding run of zero valued coefficients. The amplitudes vary from 65 to 4096. The mapping from amplitude to symbol is explained in Section 4.1.1.
8. $R = \{R_1, R_2, \ldots, R_{max}\}$: a run of 1 or more zero valued coefficients. $R_1 = 1$ and $R_{max} = 62$.
9. $E = \{EOB\}$: the end of block symbol.

Figure 7 shows how a zero run length and amplitude coefficients are mapped to the sets A^{nrb}, A^{nri}, A^{wrb}, A^{wri} and R.

4.1.1 Indexed Amplitude to Amplitude Symbol Mapping

When a coefficient's amplitude falls inside the index range, it encoded by a variable length code word and an index value. The index value, P, is computed by:

$$P = ((A-1) >> 4), \quad 65 \le A \le 4096.$$

The value used to determine the variable length code word, V, is computed according to:

$$\hat{A} = A - (P << 4) \quad 1 \le \hat{A} \le 64$$

$$V = VLCLUT(\hat{A})$$

A set of Huffman code words is generated for the symbols in the five sets of $A^{nrb}, A^{nri}, A^{wrb}, A^{wri}, E$ which results in a set of amplitude code words $V^A = \{V^{nrb}, V^{nri}, V^{wrb}, V^{wri}, V^E\}$. There are $4*64+1=129$ code words in V^A. Another set of Huffman code words is generated for 62 the symbols in R, which results in a set of zero-run code words V^R.

If a coefficient is an element in the set A^{nrb} (i.e., the dotted region of Figure 7), a single code word is inserted into the encoded symbol bit stream. The format for this case is shown in the top of Figure 8. If a coefficient resides in the set A^{wrb} two code words are inserted into the encoded symbol bit-stream: one for the coefficient taken from V^{wrb}, and the second for the number of preceding zero runs, taken from V^R. This case is depicted in the drawing that is second from the top of Figure 8. If a coefficient has no preceding run of zeros and lies in the index range, it is an element of A^{nri}. In this case, the codeword is taken from V^{nri} and the 6-bit index value P is appended onto the code word, as shown in the middle drawing of Figure 8. A coefficient that is a member of the set A^{wri} is encoded as shown in the drawing that is third from the top of Figure 8. This format uses a codeword taken from V^{wri} and a 6-bit index P to represent the amplitude and a codeword taken from V^R to represent the number of preceding zero

runs. Finally, the end of block case is shown in the bottom ofFigure 8. Here a single 4-bit code word is inserted into the encoded bit-stream.

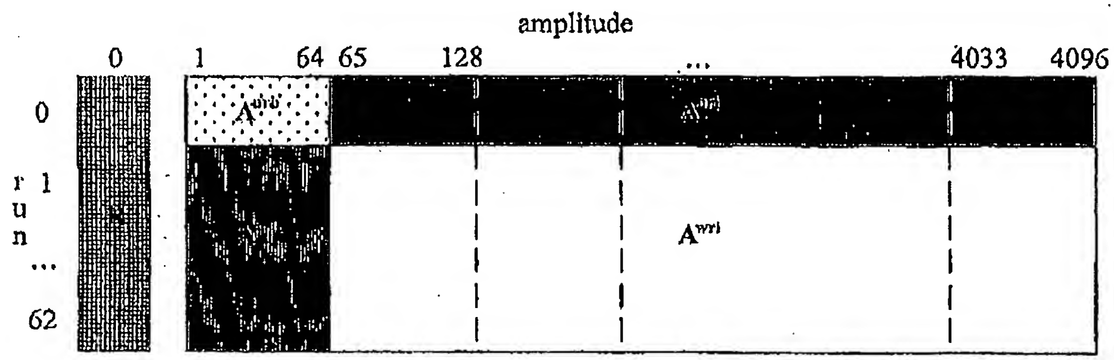

Figure 7 Mapping of Run/Amplitude Combinations to Symbol Sets